CLAIMS

1. A method for controlling a data output frequency, comprising:

defining a validity bit within a set of bits representing output data, the validity bit indicating a status of the output data;

defining a validity bit value pattern associated with a particular data output frequency;

assigning successive validity bit values of the validity bit value pattern to the validity bit in accordance with each cycle of a clock signal;

transmitting the set of bits representing output data in accordance with each cycle of the clock signal; and

processing the output data according to the validity bit.

2. A method for controlling a data output frequency as recited in claim 1, wherein the validity bit indicates the status of the output data as being one of valid and invalid.

3. A method for controlling a data output frequency as recited in claim 2, wherein the validity bit indicating the status of the output data as being valid causes the output data to be processed normally.

4. A method for controlling a data output frequency as recited in claim 2, wherein the validity bit indicating the status of the output data as being invalid causes the output data to not be processed.

5. A method for controlling a data output frequency as recited in claim 1, wherein the validity bit value pattern represents a sequence of validity bit values, the sequence of validity bit values including valid values indicating the status of the output data as being valid,

the valid values occurring within the sequence of validity bit values at a frequency corresponding to the particular data output frequency.

6. A method for controlling a data output frequency as recited in claim 1, wherein the clock signal is defined to maintain a substantially constant clock signal frequency, the clock signal frequency being an integer multiple of the particular data output frequency.

7. A method for controlling a data output frequency as recited in claim 1, further comprising:

transmitting the set of bits representing output data in accordance with each cycle of the clock signal through pipelined logic defined by a plurality of sequentially configured logic stages; and

operating each of the plurality of sequentially configured logic stages to manipulate the set of bits representing output data, whereby the validity bit within a set of bits representing output data remains unchanged.

8. A method for testing circuitry, comprising:

providing a field programmable gate array (FPGA) and an application specific integrated circuit (ASIC), wherein the FPGA and ASIC have substantially similar logic;

defining an FPGA validity bit pattern associated with a data output frequency;

defining an ASIC validity bit pattern associated with the data output frequency; and

operating the FPGA and the ASIC in accordance with their respective validity bit pattern to generate test results.

9. A method for testing circuitry as recited in claim 8, further comprising:

defining an FPGA validity bit within a set of bits representing FPGA output data, the FPGA validity bit indicating a status of the FPGA output data as being one of valid and invalid; and

defining an ASIC validity bit within a set of bits representing ASIC output data, the ASIC validity bit indicating a status of the ASIC output data as being one of valid and invalid.

10. A method for testing circuitry as recited in claim 9, further comprising:

assigning successive validity bit values of the FPGA validity bit pattern to the FPGA validity bit in accordance with each cycle of an FPGA clock signal; and

assigning successive validity bit values of the ASIC validity bit pattern to the ASIC validity bit in accordance with each cycle of an ASIC clock signal.

11. A method for testing circuitry as recited in claim 10, wherein the FPGA clock signal and the ASIC clock signal have different frequencies with respect to each other.

12. A method for testing circuitry as recited in claim 10, wherein operating the FPGA and the ASIC in accordance with their respective validity bit pattern includes,

receiving the set of bits representing FPGA output data in accordance with each cycle of the FPGA clock signal,

recognizing the FPGA validity bit as indicating the status of the FPGA output data to be valid and processing the FPGA output data,

recognizing the FPGA validity bit as indicating the status of the FPGA output data to be invalid and discarding the FPGA output data,

receiving the set of bits representing ASIC output data in accordance with each cycle of the ASIC clock signal,

recognizing the ASIC validity bit as indicating the status of the ASIC output data to be valid and processing the ASIC output data, and

recognizing the ASIC validity bit as indicating the status of the ASIC output data to be invalid and discarding the ASIC output data.

13. A method for testing circuitry as recited in claim 10, wherein a frequency of occurrence of an FPGA validity bit value indicating a valid status of the FPGA output data corresponds to the data output frequency, and

wherein a frequency of occurrence of an ASIC validity bit value indicating a valid status of the ASIC output data corresponds to the data output frequency.

14. A method for testing circuitry as recited in claim 9, further comprising:

transmitting the set of bits representing FPGA output data in accordance with each cycle of the FPGA clock signal through pipelined logic defined by a plurality of sequentially configured logic stages; and

operating each of the plurality of sequentially configured logic stages to manipulate the set of bits representing FPGA output data, whereby the FPGA validity bit within a set of bits representing FPGA output data remains unchanged.

15. A method for testing circuitry as recited in claim 9, further comprising:

transmitting the set of bits representing ASIC output data in accordance with each cycle of the ASIC clock signal through pipelined logic defined by a plurality of sequentially configured logic stages; and

operating each of the plurality of sequentially configured logic stages to manipulate the set of bits representing ASIC output data, whereby the ASIC validity bit within a set of bits representing ASIC output data remains unchanged.

16. A data output frequency control module, comprising:

a data rate control module configured to output a validity bit signal in accordance with a clock signal to be generated by a clock circuit;

a logic stage configured to receive a data signal and the validity bit signal; and

a transceiver connected to receive the data signal and the validity bit signal from the logic stage, the transceiver configured to recognize a valid state of the validity bit signal, the transceiver defined to process the data signal in response to the valid state.

17. A data output frequency control module as recited in claim 16, wherein the transceiver is further configured to recognize an invalid state of the validity bit signal, the transceiver defined to discard the data signal in response to the invalid state.

18. A data output frequency control module as recited in claim 16, wherein the data rate control module is further configured to generate a validity bit pattern representing a sequence of validity bit signals, the validity bit pattern being defined to cause the transceiver to process the data signal at a rate corresponding to a data output frequency.

19. A data output frequency control module as recited in claim 16, wherein the logic stage is connected to receive the data signal from a data buffer.

20. A data output frequency control module as recited in claim 16, wherein the logic stage is included within pipelined logic defined by a plurality of sequentially connected logic stages, wherein each of the plurality of sequentially connected logic stages is configured to manipulate the data signal without affecting the validity bit signal.